



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/565,132	03/29/2006	Sang-Won Jeong	CU-4661 WWP	9058
26530	7590	01/15/2009	EXAMINER	
LADAS & PARRY LLP 224 SOUTH MICHIGAN AVENUE SUITE 1600 CHICAGO, IL 60604			COLUCCI, MICHAEL C	
		ART UNIT	PAPER NUMBER	
		2626		
		MAIL DATE		DELIVERY MODE
		01/15/2009		PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

<b>Office Action Summary</b>	<b>Application No.</b>	<b>Applicant(s)</b>	
	10/565,132	JEONG ET AL.	
	<b>Examiner</b>	<b>Art Unit</b>	
	MICHAEL C. COLUCCI	2626	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

#### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### Status

- 1) Responsive to communication(s) filed on 13 November 2008.
- 2a) This action is **FINAL**.                    2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### Disposition of Claims

- 4) Claim(s) 1-11 is/are pending in the application.
  - 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) Claim(s) \_\_\_\_\_ is/are allowed.
- 6) Claim(s) 1-11 is/are rejected.
- 7) Claim(s) \_\_\_\_\_ is/are objected to.
- 8) Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on \_\_\_\_\_ is/are: a) accepted or b) objected to by the Examiner.
 

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
  - a) All    b) Some \* c) None of:
    1. Certified copies of the priority documents have been received.
    2. Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
    3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

#### Attachment(s)

- |  |   |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892)                     | 4) <input type="checkbox"/> Interview Summary (PTO-413)           |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ .                                    |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)          | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____ .  | 6) <input type="checkbox"/> Other: _____ .                        |

***Continued Examination Under 37 CFR 1.114***

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 11/13/2008 has been entered.

***Response to Arguments***

2. Applicant's arguments filed 11/13/2008 have been fully considered but they are not persuasive.

**Argument (page 11):**

- “The applicants can find nothing within Rust, Tokieda and Morimoto, in whole or in combination, that teaches or suggests, inter alia, the newly added limitations (or similar limitations) that require:

when the parsed rights term cannot be interpreted by the multilingual RDD registry of the central system, the central system connects to another local system to extract interpreting information of the rights term from the another local system.

Therefore, the applicants submit that the combination of Rust, Tokieda

and Tokieda cannot support an obviousness rejection to the presently claimed invention”

**Response to argument:**

Examiner takes the position that Rust alone is directly within the scope of the present invention, wherein Rust teaches a system used for translation between languages according to an interoperability of a rights data dictionary. Further, Rust like the present invention demonstrates parallel teachings pertaining to the mapping of rights data dictionary (present invention specification page 8).

Additionally, the teachings of Rust in view of Tokieda and Morimoto as a whole are also directly in parallel with the teachings of the present invention (present invention Drawings figure 3). Rust teaches well known methods of language translation, wherein Rust teaches an adaptable hierarchical dictionary-like data structure may be constructed using the context model for various applications.

The data structure may be used to interpret expressions that use its terms. It may also be used to translate from one expression language into another.

Because each term generated by the context model is unique and has a single meaning, and because the dictionary can be extended to any possible level of granularity, it is particularly suitable for mapping terms from a variety of different expression languages for translation purposes. Accordingly, the invention further provides a method for mapping between expression languages using a data dictionary ([0027]).

Tokieda has been introduced to strengthen the teachings of Rust, particularly relevant to figure 3 of the present invention (present invention Drawings figure 3), wherein Tokieda teaches multiple candidate translation languages in the form of a Web site connected to a TCP/IP transmission circuit network has been built for each of Japanese, English, German, French, and so forth in many languages. A Web server processes the respective languages (ranging from Japanese to French and so forth) in the respective Web sites. Such a Web site is a processing system of hardware having software installed in it, namely, an individual firmware (Tokieda [0004]). The teachings of both Rust and Tokieda are well known, wherein input information during translation is compared to several languages until a match is found, such as the case where a main network location compares to other sources until a match is found.

Furthermore, Tokieda teaches a system where various forms of interpretation are present when one system is not capable of interpreting/translating, the system moves on to another system, wherein Tokieda clearly appears to teach a general-purpose Web browser which the multilingual translation Web site apparatus 2 is provided with cannot read such a language on occasion. Therefore, a Web browser of an application capable of reading many languages is installed. For example, an application for performing judgment by tracking a language possible to be transferred is installed in advance. This tracking is to compare a part of a transferred character string with character strings of many

languages stored in advance. And like "cookie", its language is judged by identifying that it is a former visitor to the home page. It is acceptable to identify a navigation language or navigation browser language returned by the installed Web browser. Next, the multilingual translation Web site apparatus 2 transfers a manuscript for translation requested together with the time limit of delivery and special notes to a translator apparatus 3 side capable of performing translation between languages specified by the translation request. This is performed through a "multilingual translation network" in which translators performing translation in many languages are organized on the Internet. Next, the translator apparatus 3 side performs translation between the specified languages. In this case, a translator installs a translation software into a general-purpose small computer of the translator apparatus 3 and performs its machine translation, or performs a manual input translation by means of a word processor software and makes a translation writing by its electronic data and transfers it together with a bill to the multilingual translation Web site apparatus 2. In case that a translator apparatus 3 performs an ;automatic machine translation, the translator apparatus 3 performs a translation application of the multilingual translation Web site apparatus 2 through CGI or installs a translation application in it and performs the translation application in a stand-alone state. And the multilingual translation Web site apparatus 2 side may perform an automatic machine translation and then transfer its translation writing to a translator apparatus 3, and may have a translator brush up the translation including examination ([0106]- [0110] & Fig. 3).

Finally, Morimoto clearly demonstrates the back and forth ability of translation of languages through multiple dictionaries, wherein Morimoto teaches that if a main dictionary is not capable of interpreting data, other dictionaries are consulted.

Thus, Morimoto teaches that if the dictionary entry does not exist in its own server dictionary, the dictionary server according to the present invention issues a dictionary entry retrieval request to other dictionary server so that the dictionary server according to the present invention may use dictionaries distributed on the network. At that time, if it is clear which dictionary server the retrieved dictionary entry exists, i.e. the dictionary entry exists in the distributed dictionary index 9, the dictionary server issues the retrieval request to such dictionary server.

However, if the dictionary server having dictionary information of dictionary entry is not clear, i.e. the dictionary entry does not exist in the distributed dictionary index 9, then the dictionary server according to the present invention makes an inquiry to other dictionary servers in the dictionary server list 6. At that time, when the dictionary server makes an inquiry to an arbitrary dictionary server in the dictionary server list 6, a retrieval efficiency is considerably low. Also, when each dictionary server makes an inquiry to a plurality of dictionary servers at the same time, a traffic increases exponentially, which causes a serious problem. Therefore, it is prepared that the dictionary server according to this embodiment is provided with a function to determine a priority of a dictionary server when a

document which is a translation target is translated, and makes an inquiry in the sequential order of high priority (Morimoto Col. 12 lines 1-25).

***Claim Rejections - 35 USC § 103***

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claim 1-11 rejected under 35 U.S.C. 103(a) as being unpatentable over Rust et al. USPGPUB 20030221171 A1 (hereinafter Rust) in view of Tokieda et al. USPGPUB 20020152063 A1 (hereinafter Tokieda) and further in view of Morimoto et al. US 6789057 B1 (hereinafter Morimoto).

Re claims 1 and 7, Rust teaches a network of local systems, for connecting to a central system, the network comprising:

the central system comprising:

a multilingual rights data dictionary (RDD) ([0115]) registry storing a multilingual RDD

a local RDD registry for storing an RDD ([0114] – [0115] & Tables V, VI) of a specific language ([0066]);

a processing means for parsing a rights term ([0038] & Fig. 10) and interpreting the rights term by referring to the local RDD registry ([0004]),

wherein the processing means acquires rights term interpreting information ([0004]) based on the multilingual RDD registry ([0114] – [0115] & Tables V, VI)

However, Rust fails to teach connecting to the central system when the rights term interpreting information does not exist in the local RDD registry ([0153] & Fig. 5)

Tokieda teaches the multilingual processing database is provided with a plurality of servers, which can be distributively disposed. The database acquires an actual server name, server position, multilingual processing database name and the like from database management data in response to a request for a language ID, page ID and the like, and accesses a multilingual processing database desired out of database servers. Additionally, Tokieda teaches a master web site containing language data for multilingual translation as well as external multilingual processing databases containing several languages Japanese, German, English, French, etc., that can be referenced when non-language (image) or language is not handled through the master website (Tokieda [0007] – [0008] & Fig. 5).

Further, Tokieda teaches a quick and inexpensive translation service in many languages by means of a single Web site (firmware) on a communication network. In order to perform a quick translation in many languages requested through a communication network, the invention has been composed so that a multilingual translation Web site apparatus 2 functioning as a Web site (firmware) consisting of one apparatus and one translation processing system receives language data of a subject of translation from a translation requester apparatus 4. The multilingual translation Web site apparatus 2 changes its processing form adaptively to the language of the subject

Art Unit: 2626

of translation received, and automatically selects language data for translation. A translator apparatus 3 performs translation of the language data received from the multilingual translation Web site apparatus 2. The multilingual translation Web site apparatus 2 receives and enters the translated data from the translator apparatus 3 into the multilingual processing database, and automatically changes its processing form of translation adaptively to the language after translation, and enables the translation requester apparatus 4 to receive the translated data (Tokieda Abstract).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust to incorporate a multilingual rights data dictionary that connects to a central system when local references do not exist as taught by Tokieda to allow for outsourcing a database (such as through the internet) allows for increased language databases, where the local language data does not need to contain all language for translation and having external databases provides a faster method of translation, where the original RDD data can create genealogies, context description, and logical relationships based on a foreign language (Tokieda Abstract).

However, Rust in view of Tokieda fails to teach a local RDD registry storing a specific language RDD

a network of local systems

wherein when the rights term cannot be interpreted by the local RDD registry of a particular local system, the particular local system acquires the rights term interpreting information based on the multilingual RDD registry by connecting to the central system to interpret the rights term

**NOTE:** When read in light of the specification and Fig. 3 of the present invention, a network of local systems is construed to be functionally equivalent and equally effective to that of multiple dictionaries linked together.

Morimoto teaches an example in which users share a machine translation dictionary by effectively utilizing a network, and proposes a system which is connected to the network via dictionaries. When this system is applied to a translation, translation dictionaries of respective languages such as English and Russian are previously connected to the system and a language of an inputted word is discriminated. Thereafter, inputted data (image data of character) is transferred to a center having a corresponding translation dictionary. According to the above-mentioned related art, dictionaries that are used to translate specific fields or specific languages are limited, and words that could not be translated such dictionaries are processed as words which cannot be translated (Morimoto Col. 1 lines 21-34).

Morimoto further improves this limitation by teaching that it is determined by other dictionary server at the next decision step S15 whether or not transmitted registered dictionary entry information is received. If the transmitted registered dictionary entry information is received as represented by a YES at the decision step S15, then other dictionary server executes a dictionary entry registration processing (step S16). Specifically, the dictionary server compares the received new dictionary entry information with the dictionary. If it is determined that the new dictionary entry is an unknown word, then a dictionary entry and the ID number of the server in the new dictionary entry information are stored in the distributed dictionary index 9. In this

manner, each dictionary server is able to learn dictionary entry information of dictionaries of a plurality of other dictionary servers distributed and connected to the system. At that time, for the dictionary server which receives the new dictionary entry information, new dictionary entry information is only the dictionary entry and the IP address of the internet protocol. Thus, as compared with the case in which new dictionary entry information containing equivalents, a discrimination of a part of speech, a semantic or syntactic attribute and a field of a dictionary entry is registered, it is frequently observed that the dictionary server may save a dictionary capacity. Moreover, the dictionary server which transmits new dictionary entry information is able to prevent information of equivalent, which is newly registered by the above dictionary server, from being duplicated without permission. Thus, the present invention may be applied to the case in which an accounting for communicating an equivalent, a part of speech, an attribute and a field is executed between the dictionary servers (Morimoto Col. 5 lines 65 – Col. 7 line 29).

Morimoto teaches that if the dictionary entry does not exist in its own server dictionary, the dictionary server according to the present invention issues a dictionary entry retrieval request to other dictionary server so that the dictionary server according to the present invention may use dictionaries distributed on the network. At that time, if it is clear which dictionary server the retrieved dictionary entry exists, i.e. the dictionary entry exists in the distributed dictionary index 9, the dictionary server issues the retrieval request to such dictionary server. However, if the dictionary server having dictionary information of dictionary entry is not clear, i.e. the dictionary entry does not exist in the

distributed dictionary index 9, then the dictionary server according to the present invention makes an inquiry to other dictionary servers in the dictionary server list 6. At that time, when the dictionary server makes an inquiry to an arbitrary dictionary server in the dictionary server list 6, a retrieval efficiency is considerably low. Also, when each dictionary server makes an inquiry to a plurality of dictionary servers at the same time, a traffic increases exponentially, which causes a serious problem. Therefore, it is prepared that the dictionary server according to this embodiment is provided with a function to determine a priority of a dictionary server when a document which is a translation target is translated, and makes an inquiry in the sequential order of high priority (Morimoto Col. 12 lines 1-25).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust in view of Tokieda to incorporate a local RDD registry storing a specific language RDD and a network of local systems, wherein when the rights term cannot be interpreted by the local RDD registry of a particular local system, the particular local system acquires the rights term interpreting information based on the multilingual RDD registry by connecting to the central system to interpret the rights term and when the parsed rights term cannot be interpreted by the multilingual RDD registry of the central system the central system connects to another local system to extract interpreting information from the another local system of the parsed rights term as taught by Morimoto to allow for a group of dictionaries that communicate to enable a network that can be updated, wherein various languages are translated in the appropriate language by checking the input, updating the all

dictionaries, and reducing memory by efficiently not repeating the translation of text (Morimoto Col. 5 lines 65 – Col. 7 line 29) and also to allow for an updatable and recursive dictionary search routine that can start and end with the initial dictionary in a network of multiple dictionaries giving multiple languages (Morimoto Col. 12 lines 1-25).

Re claims 2, 5, 8, and 11, Rust teaches the network as recited in claim 1, wherein the multilingual RDD registry of the central system comprises:

an aggregate set of the specific language RDD's ([0114] – [0115] & Tables V, VI) stored in the local systems

wherein the rights term interpreting information is extracted ([0004])

a link for connecting the local RDDs logically ([0101]),

However, Rust fails to teach based on the link a plurality of local systems (Tokieda [0153] & Fig. 5)

Tokieda teaches the multilingual processing database is provided with a plurality of servers, which can be distributively disposed. The database acquires an actual server name, server position, multilingual processing database name and the like from database management data in response to a request for a language ID, page ID and the like, and accesses a multilingual processing database desired out of database servers. Additionally, Tokieda teaches a master web site containing language data for multilingual translation as well as external multilingual processing databases containing several languages Japanese, German, English, French, etc., that can be referenced when non-language (image) or language is not handled through the master website.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust to incorporate a multilingual rights data dictionary that connects to a central system when local references do not exist as taught by Tokieda to allow for outsourcing a database (such as through the internet) allows for increased language databases, where the local language data does not need to contain all language for translation and having external databases provides a faster method of translation, where the original RDD data can create genealogies, context description, and logical relationships based on a foreign language (Tokieda Abstract).

However, Rust in view of Tokieda fails to teach a local RDD registry storing a specific language RDD

a network of local systems

**NOTE:** When read in light of the specification and Fig. 3 of the present invention, a network of local systems is construed to be functionally equivalent and equally effective to that of multiple dictionaries linked together.

Morimoto teaches an example in which users share a machine translation dictionary by effectively utilizing a network, and proposes a system which is connected to the network via dictionaries. When this system is applied to a translation, translation dictionaries of respective languages such as English and Russian are previously connected to the system and a language of an inputted word is discriminated. Thereafter, inputted data (image data of character) is transferred to a center having a corresponding translation dictionary. According to the above-mentioned related art, dictionaries that are used to translate specific fields or specific languages are limited,

and words that could not be translated such dictionaries are processed as words which cannot be translated (Morimoto Col. 1 lines 21-34).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust in view of Tokieda to incorporate a local RDD registry storing a specific language RDD and a network of local systems as taught by Morimoto to allow for a system of translation dictionaries that are transferred to a main center, wherein a specific dictionary is chosen based on language and unrecognizable words are identified (Morimoto Col. 1 lines 21-34).

Re claims 3 and 9, Rust teaches the local system as recited in claim I, wherein the processing means acquires the rights term interpreting information ([0004]) from the local RDD registry ([0114] – [0115] & Tables V, VI)

However, Rust fails to teach another local system linked to the multilingual RDD registry (Tokieda [0153] & Fig. 5)

Tokieda teaches the multilingual processing database is provided with a plurality of servers, which can be distributively disposed. The database acquires an actual server name, server position, multilingual processing database name and the like from database management data in response to a request for a language ID, page ID and the like, and accesses a multilingual processing database desired out of database servers. Additionally, Tokieda teaches a master web site containing language data for multilingual translation as well as external multilingual processing databases containing

several languages Japanese, German, English, French, etc., that can be referenced when non-language (image) or language is not handled through the master website.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust to incorporate a multilingual rights data dictionary that connects to a central system when local references do not exist as taught by Tokieda to allow for outsourcing a database (such as through the internet) allows for increased language databases, where the local language data does not need to contain all language for translation and having external databases provides a faster method of translation, where the original RDD data can create genealogies, context description, and logical relationships based on a foreign language (Tokieda Abstract).

However, Rust in view of Tokieda fails to teach a local RDD registry storing a specific language RDD

a network of local systems

when the parsed rights term cannot be interpreted by the multilingual RDD registry of the central system the central system connects to another local system to extract interpreting information from the another local system of the parsed rights term

**NOTE:** When read in light of the specification and Fig. 3 of the present invention, a network of local systems is construed to be functionally equivalent and equally effective to that of multiple dictionaries linked together.

Morimoto teaches an example in which users share a machine translation dictionary by effectively utilizing a network, and proposes a system which is connected to the network via dictionaries. When this system is applied to a translation, translation

dictionaries of respective languages such as English and Russian are previously connected to the system and a language of an inputted word is discriminated. Thereafter, inputted data (image data of character) is transferred to a center having a corresponding translation dictionary. According to the above-mentioned related art, dictionaries that are used to translate specific fields or specific languages are limited, and words that could not be translated such dictionaries are processed as words which cannot be translated (Morimoto Col. 1 lines 21-34).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust in view of Tokieda to incorporate a local RDD registry storing a specific language RDD and a network of local systems as taught by Morimoto to allow for a system of translation dictionaries that are transferred to a main center, wherein a specific dictionary is chosen based on language and unrecognizable words are identified (Morimoto Col. 1 lines 21-34).

Re claims 4 and 10, Rust teaches a network of a central system of a multilingual rights data dictionary (RDD) ([0109]), the network comprising:  
a multilingual RDD registry ([0114] – [0115] & Tables V, VI);  
a central system processing means for receiving a rights term from the connected local system, extracting interpreting information of the rights term ([0004]) based on the multilingual RDD registry and transmitting the interpreting information ([0114] – [0115] & Tables V, VI)

However, Rust fails to teach for connecting to a plurality of local systems

(Tokieda [0153] & Fig. 5)

Tokieda teaches the multilingual processing database is provided with a plurality of servers, which can be distributively disposed. The database acquires an actual server name, server position, multilingual processing database name and the like from database management data in response to a request for a language ID, page ID and the like, and accesses a multilingual processing database desired out of database servers. Additionally, Tokieda teaches a master web site containing language data for multilingual translation as well as external multilingual processing databases containing several languages Japanese, German, English, French, etc., that can be referenced when non-language (image) or language is not handled through the master website.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust to incorporate a multilingual rights data dictionary that connects to a central system when local references do not exist as taught by Tokieda to allow for outsourcing a database (such as through the internet) allows for increased language databases, where the local language data does not need to contain all language for translation and having external databases provides a faster method of translation, where the original RDD data can create genealogies, context description, and logical relationships based on a foreign language (Tokieda Abstract).

However, Rust in view of Tokieda fails to teach a local RDD registry storing a specific language RDD

a network of local systems

wherein when the rights term cannot be interpreted by the local RDD registry of a particular local system, the particular local system acquires the rights term interpreting information based on the multilingual RDD registry by connecting to the central system to interpret the rights term

when the parsed rights term cannot be interpreted by the multilingual RDD registry of the central system the central system connects to another local system to extract interpreting information from the another local system of the parsed rights term

**NOTE:** When read in light of the specification and Fig. 3 of the present invention, a network of local systems is construed to be functionally equivalent and equally effective to that of multiple dictionaries linked together.

Morimoto teaches an example in which users share a machine translation dictionary by effectively utilizing a network, and proposes a system which is connected to the network via dictionaries. When this system is applied to a translation, translation dictionaries of respective languages such as English and Russian are previously connected to the system and a language of an inputted word is discriminated. Thereafter, inputted data (image data of character) is transferred to a center having a corresponding translation dictionary. According to the above-mentioned related art, dictionaries that are used to translate specific fields or specific languages are limited, and words that could not be translated such dictionaries are processed as words which cannot be translated (Morimoto Col. 1 lines 21-34).

Morimoto further improves this limitation by teaching that it is determined by other dictionary server at the next decision step S15 whether or not transmitted

registered dictionary entry information is received. If the transmitted registered dictionary entry information is received as represented by a YES at the decision step S15, then other dictionary server executes a dictionary entry registration processing (step S16). Specifically, the dictionary server compares the received new dictionary entry information with the dictionary. If it is determined that the new dictionary entry is an unknown word, then a dictionary entry and the ID number of the server in the new dictionary entry information are stored in the distributed dictionary index 9. In this manner, each dictionary server is able to learn dictionary entry information of dictionaries of a plurality of other dictionary servers distributed and connected to the system. At that time, for the dictionary server which receives the new dictionary entry information, new dictionary entry information is only the dictionary entry and the IP address of the internet protocol. Thus, as compared with the case in which new dictionary entry information containing equivalents, a discrimination of a part of speech, a semantic or syntactic attribute and a field of a dictionary entry is registered, it is frequently observed that the dictionary server may save a dictionary capacity. Moreover, the dictionary server which transmits new dictionary entry information is able to prevent information of equivalent, which is newly registered by the above dictionary server, from being duplicated without permission. Thus, the present invention may be applied to the case in which an accounting for communicating an equivalent, a part of speech, an attribute and a field is executed between the dictionary servers (Morimoto Col. 5 lines 65 – Col. 7 line 29).

Morimoto teaches that if the dictionary entry does not exist in its own server dictionary, the dictionary server according to the present invention issues a dictionary entry retrieval request to other dictionary server so that the dictionary server according to the present invention may use dictionaries distributed on the network. At that time, if it is clear which dictionary server the retrieved dictionary entry exists, i.e. the dictionary entry exists in the distributed dictionary index 9, the dictionary server issues the retrieval request to such dictionary server. However, if the dictionary server having dictionary information of dictionary entry is not clear, i.e. the dictionary entry does not exist in the distributed dictionary index 9, then the dictionary server according to the present invention makes an inquiry to other dictionary servers in the dictionary server list 6. At that time, when the dictionary server makes an inquiry to an arbitrary dictionary server in the dictionary server list 6, a retrieval efficiency is considerably low. Also, when each dictionary server makes an inquiry to a plurality of dictionary servers at the same time, a traffic increases exponentially, which causes a serious problem. Therefore, it is prepared that the dictionary server according to this embodiment is provided with a function to determine a priority of a dictionary server when a document which is a translation target is translated, and makes an inquiry in the sequential order of high priority (Morimoto Col. 12 lines 1-25).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust in view of Tokieda to incorporate a local RDD registry storing a specific language RDD and a network of local systems, wherein when the rights term cannot be interpreted by the local RDD registry of a

particular local system, the particular local system acquires the rights term interpreting information based on the multilingual RDD registry by connecting to the central system to interpret the rights term and when the parsed rights term cannot be interpreted by the multilingual RDD registry of the central system the central system connects to another local system to extract interpreting information from the another local system of the parsed rights term as taught by Morimoto to allow for a group of dictionaries that communicate to enable a network that can be updated, wherein various languages are translated in the appropriate language by checking the input, updating the all dictionaries, and reducing memory by efficiently not repeating the translation of text (Morimoto Col. 5 lines 65 – Col. 7 line 29) and also to allow for an updatable and recursive dictionary search routine that can start and end with the initial dictionary in a network of multiple dictionaries giving multiple languages (Morimoto Col. 12 lines 1-25).

Re claim 6, Rust teaches a network of a central system of a multilingual rights data dictionary (RDD) ([0109]), the network comprising:

a multilingual RDD registry for including link information ([0114] – [0115] & Tables V, VI) which logically connects local RDD registries maintained by the plurality of local systems ([0101]);

a processing means for receiving a rights term from the connected local system and transmitting information which is needed to interpret the rights ([0004]) term based on the multilingual RDD registry to the local system ([0114] – [0115] & Tables V, VI)

for connecting to a plurality of local systems (Tokieda [0153] & Fig. 5)

Tokieda teaches the multilingual processing database is provided with a plurality of servers, which can be distributively disposed. The database acquires an actual server name, server position, multilingual processing database name and the like from database management data in response to a request for a language ID, page ID and the like, and accesses a multilingual processing database desired out of database servers. Additionally, Tokieda teaches a master web site containing language data for multilingual translation as well as external multilingual processing databases containing several languages Japanese, German, English, French, etc., that can be referenced when non-language (image) or language is not handled through the master website.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust to incorporate a multilingual rights data dictionary that connects to a central system when local references do not exist as taught by Tokieda to allow for outsourcing a database (such as through the internet) allows for increased language databases, where the local language data does not need to contain all language for translation and having external databases provides a faster method of translation, where the original RDD data can create genealogies, context description, and logical relationships based on a foreign language (Tokieda Abstract).

However, Rust in view of Tokieda fails to teach a local RDD registry storing a specific language RDD

a network of local systems

wherein when the rights term cannot be interpreted by the local RDD registry of a particular local system, the particular local system acquires the rights term interpreting

information based on the multilingual RDD registry by connecting to the central system to interpret the rights term

when the parsed rights term cannot be interpreted by the multilingual RDD registry of the central system the central system connects to another local system to extract interpreting information from the another local system of the parsed rights term

**NOTE:** When read in light of the specification and Fig. 3 of the present invention, a network of local systems is construed to be functionally equivalent and equally effective to that of multiple dictionaries linked together.

Morimoto teaches an example in which users share a machine translation dictionary by effectively utilizing a network, and proposes a system which is connected to the network via dictionaries. When this system is applied to a translation, translation dictionaries of respective languages such as English and Russian are previously connected to the system and a language of an inputted word is discriminated. Thereafter, inputted data (image data of character) is transferred to a center having a corresponding translation dictionary. According to the above-mentioned related art, dictionaries that are used to translate specific fields or specific languages are limited, and words that could not be translated such dictionaries are processed as words which cannot be translated (Morimoto Col. 1 lines 21-34).

Morimoto further improves this limitation by teaching that it is determined by other dictionary server at the next decision step S15 whether or not transmitted registered dictionary entry information is received. If the transmitted registered dictionary entry information is received as represented by a YES at the decision step

S15, then other dictionary server executes a dictionary entry registration processing (step S16). Specifically, the dictionary server compares the received new dictionary entry information with the dictionary. If it is determined that the new dictionary entry is an unknown word, then a dictionary entry and the ID number of the server in the new dictionary entry information are stored in the distributed dictionary index 9. In this manner, each dictionary server is able to learn dictionary entry information of dictionaries of a plurality of other dictionary servers distributed and connected to the system. At that time, for the dictionary server which receives the new dictionary entry information, new dictionary entry information is only the dictionary entry and the IP address of the internet protocol. Thus, as compared with the case in which new dictionary entry information containing equivalents, a discrimination of a part of speech, a semantic or syntactic attribute and a field of a dictionary entry is registered, it is frequently observed that the dictionary server may save a dictionary capacity. Moreover, the dictionary server which transmits new dictionary entry information is able to prevent information of equivalent, which is newly registered by the above dictionary server, from being duplicated without permission. Thus, the present invention may be applied to the case in which an accounting for communicating an equivalent, a part of speech, an attribute and a field is executed between the dictionary servers (Morimoto Col. 5 lines 65 – Col. 7 line 29).

Morimoto teaches that if the dictionary entry does not exist in its own server dictionary, the dictionary server according to the present invention issues a dictionary entry retrieval request to other dictionary server so that the dictionary server according

to the present invention may use dictionaries distributed on the network. At that time, if it is clear which dictionary server the retrieved dictionary entry exists, i.e. the dictionary entry exists in the distributed dictionary index 9, the dictionary server issues the retrieval request to such dictionary server. However, if the dictionary server having dictionary information of dictionary entry is not clear, i.e. the dictionary entry does not exist in the distributed dictionary index 9, then the dictionary server according to the present invention makes an inquiry to other dictionary servers in the dictionary server list 6. At that time, when the dictionary server makes an inquiry to an arbitrary dictionary server in the dictionary server list 6, a retrieval efficiency is considerably low. Also, when each dictionary server makes an inquiry to a plurality of dictionary servers at the same time, a traffic increases exponentially, which causes a serious problem. Therefore, it is prepared that the dictionary server according to this embodiment is provided with a function to determine a priority of a dictionary server when a document which is a translation target is translated, and makes an inquiry in the sequential order of high priority (Morimoto Col. 12 lines 1-25).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the system of Rust in view of Tokieda to incorporate a local RDD registry storing a specific language RDD and a network of local systems, wherein when the rights term cannot be interpreted by the local RDD registry of a particular local system, the particular local system acquires the rights term interpreting information based on the multilingual RDD registry by connecting to the central system to interpret the rights term and when the parsed rights term cannot be interpreted by the

multilingual RDD registry of the central system the central system connects to another local system to extract interpreting information from the another local system of the parsed rights term as taught by Morimoto to allow for a group of dictionaries that communicate to enable a network that can be updated, wherein various languages are translated in the appropriate language by checking the input, updating the all dictionaries, and reducing memory by efficiently not repeating the translation of text (Morimoto Col. 5 lines 65 – Col. 7 line 29) and also to allow for an updatable and recursive dictionary search routine that can start and end with the initial dictionary in a network of multiple dictionaries giving multiple languages (Morimoto Col. 12 lines 1-25).

### ***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael C. Colucci whose telephone number is (571)-270-1847. The examiner can normally be reached on 9:30 am - 6:00 pm, Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Richemond Dorvil can be reached on (571)-272-7602. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Michael C Colucci/  
Examiner, Art Unit 2626  
Patent Examiner  
AU 2626  
(571)-270-1847  
[Michael.Colucci@uspto.gov](mailto:Michael.Colucci@uspto.gov)

/Richemond Dorvil/  
Supervisory Patent Examiner, Art Unit 2626